IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re U.S. Patent Application of:	)	**Group Art Unit: 2191**
	)	
Detlev Wittmer *et al.*	)	**Examiner: Junchun Wu**
	)	
Serial Number: 10/534,700	)	***Attorney Docket*: WITT3005/FJD**
	)	
Filed: April 25, 2006	)	**Confirmation No.: 8701**

For: Method for Producing Software Modules for Field Appliances Used in the Process Automation Technique

APPELLANT'S BRIEF UNDER 37 C.F.R. §1.192

Sir:

This paper is in furtherance of the Notice of Appeal filed in this case on February 14, 2011. The fee required under 37 C.F.R. §1.17(f) accompanies this Appeal Brief.

This Brief contains these items under the following headings and in the order set forth below:

I. Real Party In Interest

The real party in interest is Endress+Hauser Conducta Gesellschaft für Mess- u. Regeltechnik mbH+Co. KG.

II. Related Appeals And Interferences

There are no related appeals or interferences.

III. Status of Claims

The status of the claims in this application is:

A. Status of all the claims

1. Claims canceled: 1-4
2. Claims withdrawn from consideration: None
3. Claims pending: 5-10
4. Claims allowed: None
5. Claims objected to: None
6. Claims rejected: 5-10

B. Claims on Appeal:

The claims on appeal are: 5-10.

IV. Status of Amendments

An amendment under 37 CFR 1.116 to correct informalities in claims 8 and 9 is submitted herewith.

No other amendments have been submitted subsequent to the final rejection mailed November 24, 2010.

V. Summary of Claimed Subject Matter (with page and line references to the original English language specification)

The claimed subject matter addresses the problem that it is difficult to configure different field devices in an automated plant because the field devices use different configuration programs with different interfaces and protocols **[page 1, lines 22-30 and page 2, lines 1-10]**. Attempts have been made to provide software modules that that provide a common user interface in order to simplify configuration of different field devices, but only a few manufacturers have adapted their field devices to include such software modules **[page 2, lines 11-31]**. To solve this problem, the standard software modules can be compiled from device descriptions provided with the field devices **[page 2, lines 31-35]**. However, while this enables a standardized software module to be utilized, the device descriptions do not currently exist in a common form or language. As a result, it is necessary to provide different compilers for each different type of device description. Furthermore, in case of changes in device description protocols, corresponding revisions must be made to the compiler **[page 3, lines 1-13]**.

The present invention also provides a compiler for compiling device descriptions into standard configuration software modules, but instead of providing a different compiler for each type of device description, the claimed invention adds a device **[generator G1 or compiler C1 shown in Fig. 2 and described in lines 23-29 on page 4]** that is dedicated to generating syntactically correct standard device descriptions from various different types of device descriptions. The syntactically correct standard device descriptions are then converted into software modules by a second compiler **[compiler C, software module SM, and standard device description EDD 1.1 illustrated in Fig. 2 and described in lines 1-11 on page 5]**. The first "compiler or generator" **[G1 or C1]** in effect translates or converts the original device description of a field device **[F1, F2, F3, etc. shown in Fig. 1 and described in lines 5-8 on page 4]** into a standard device description **[EDD 1.1]** for compilation by the standard second compiler **[C]**, thereby eliminating the need to provide an entirely different compiler for each type of device description.

More specifically, the resulting method is recited in **claim 5**, as follows:

Claim 5	Support in Specification/Drawing
A method for producing software modules for field devices for process automation technology, that encapsulate all the data and functions of the corresponding field devices, wherein the software modules serve as device descriptions and have defined interfaces for application programs in process control systems, comprising the steps of:	Software modules SM are described in lines 1-11 on page 5, field devices F1, F2, F3, *etc.* are described in line 5-13 on page 4
generating syntactically correct standard device descriptions from PDM device descriptions, HCF device descriptions or company specific electronic device descriptions for field devices not having a uniform form, or language by means of a first compiler or generator; and	This step is specifically described in lines 23-29 on page 4.
converting the syntactically and semantically correct standard device descriptions further into corresponding software modules by means of a second compiler.	This step is specifically described in lines 1-8 on page 5.

Turning to the dependent claims, **claim 6** recites that interfaces and the software modules meed FDT/DTM specifications, as described in lines 25-29 on page 2, lines 14-16 on page 4 ("Profibus" corresponds to FDT/DTM), and line 3 on page 5.

Claim 7 recites different types of device descriptions that may be converted into the standard device descriptions, including PDM, HCF, and company-specific device descriptions as described in lines 23-29 on page 4.

Claim 8 recites that the standard device descriptions are EDD 1.1 device descriptions, as described in line 29 on page 4. It is noted that, due to a typographic error, a portion of claim 8 was inadvertently omitted in the last response (although it was present in the previous response). The attached amendment corrects this error.

Claim 9 recites that the EDD 1.1 device descriptions are produced from PDM device descriptions, as described in lines 23-29 on page 4 (again, a typographic error has been corrected by adding the word "from").

Finally, **claim 10** recites the second compiler produces graphical user interfaces in XML language from the EDD 1.1 device descriptions, as described in lines 12-18 on page 5.

VI. Grounds of Rejection to be Reviewed on Appeal

The sole rejection to be reviewed on appeal is a rejection of the subject matter of claims 5-10 under 35 USC §103(a) as being unpatentable in view of the publication entitled "PROFIBUS technology and application – system description," Oct. 2002, hereinafter "**PROFIBUS**," in view of Diedrich et al. ("Field Device Integration in DCS Engineering using a Device Model"), hereinafter "**Diedrich**," and further in view of Pöschmann et al. ("Experience with formal methods

implementing the PROFIBUS FMS and DP protocol for industrial applications"), herein after "**Poschmann**."

VII. Arguments

Reversal of the rejection of each of claims 5-10 under 35 USC §103(a) is respectfully requested on the grounds that the PROFIBUS, Diedrich, and Poschmann publications all fail to disclose, whether considered individually or in any reasonable combination:

- use of **two** compilers to transform different types of device descriptions (PDM, HCF, or company-specific) into "standard device descriptions" and then into "corresponding software modules" in a two-step process.

In particular, the three references fail to disclose or suggest the step of transforming one type of device description (PDM, HCF, or company-specific) into another type of device description (standard). The one reference that is alleged to disclose the claimed device description transformation, namely Diedrich, actually merely discloses generation of a compilable device description in the first place, rather than transformation of the device description from a specific type (PDM, HCF, or company-specific) into a standard type.

The PROFIBUS publication merely discloses the concept of "standardization of device management," which corresponds to the use of "special software modules" described in the background section (page 2) of Applicant's own specification. According to the method described in the PROFIBUS publication, a different compiler is needed for each type of device description. The Examiner acknowledges as much in the middle of page 3 of the Official Action.

However, the Examiner alleges that Diedrich discloses generating standard device descriptions by means of a first compiler or generator, citing page 167 of Diedrich, which reads:

> *There are two steps within the device description technology. Firstly, the device description has to be generated. This is done by compilers or generators, which translate the ASCII device description.*

The Examiner will note that this statement does not say that one device description is transformed into another type of device description. Furthermore, this passage does not mention either PDM, HCF, or company-specific device descriptions, or describe generation of a "standard" device description from a PDM, HCF, or company-specific device description. Instead, Diedrich merely discloses that a particular device description has to be generated from the ASCII device description. An ASCII "device description" is of course not a compilable description, which is why transformation is necessary. However, this does not mean that the resulting device description is a standard device description. Instead, it could very well be a company-specific device description.

Essentially, the Diedrich publication merely discloses that a device description has to be generated in order to implement PROFIBUS. This is simply an initial step in the compilation procedure described in the PROFIBUS publication, and not a modification thereof. **What Diedrich does not disclose is that a device description of one particular type (whether DCM, HCL, or other device description) is transformed by a first compiler into a *standard* device description. Instead, Diedrich merely discloses that some sort of device description has to be generated in the first place**. The device description thus generated is then compiled into a DCOM server format (which actually can be considered a standard software module). There is absolutely no suggestion in Diedrich that the compiler is not just a compiler of the type described in the PROFIBUS publication, *i.e.*, a compiler that is specific to a particular device description type.

The Poschmann publication also does not disclose a two compiler procedure, in which the first step is not merely generation of a device description, but transformation of a device description from one type to another. To the contrary, while Poschmann might disclose generation of a standard software module, generation of a standard software module is already disclosed by PROFIBUS. Thus, Poschmann does not add any relevant teachings to the disclosures of PROFIBUS and Diedrich. Both Poschmann and Diedrich describe specific implementations of PROFIBUS, in which

a different compiler would be required for PCM device descriptions, HCF device descriptions, or company specific electronic device descriptions.

Even in the claimed invention, there needs to be a step of initially generating the DCM, HCL, or other company-specific device description. This step is implied rather than being specifically disclosed, since it is inherent that the DCM, HCL, or company-specific device description must exist, and therefore have been generated, before it can be converted into a standard device description as claimed. Diedrich discloses nothing more than the step of initially generating a device description DD. The device description DD is not required to be "standard," nor is there any suggestion of subjecting the device description to a generator or compiler that transforms the device description into a "standard" device description, as opposed to a PDM, HCF, or company-specific device description that can be compiled by a second compiler into a "corresponding software module," as claimed.

The fact that Diedrich discloses translation of ASCII into a device description, whereas the transformation carried out by the claimed invention involves, in a specific embodiment, PDM to EDD 1.1, should indicate that the claimed invention involves a substantively different type of conversion or transformation than the one disclosed by Diedrich. As recited by way of example in claim 9, the invention in fact transforms the device description from one specific type (PDM) to another type (EDD 1.1) which can be used as a standard device description. The key is that one perfectly useful type of device description (PDM) is converted into another type of device description. In the prior art, when one obtains a perfectly useful type of device description, no further conversion is required. There is no reason for such a further conversion (or generation or compilation). The only reason for such a further conversion would be to provide a standardized device description to simplify the further step of compiling the device description into a corresponding software interface. However, such standardization is suggested only by Applicant's own disclosure, and not by PROFIBUS, Diedrich, or Poschmann, and therefore the subject matter

of claims 5-10 is not obvious over the proposed combination of the PROFIBUS, Diedrich, or Poschmann articles.

Conclusion

For all of the foregoing reasons, Appellants respectfully submit that the Examiner's final rejection of claims 5-10 under 35 U.S.C. §103(a) is improper and should be reversed by this Honorable Board.

Respectfully submitted,

BACON & THOMAS, PLLC

/Benjamin E. Urcia/

By: BENJAMIN E. URCIA
Registration No. 33,805

Date: April 14, 2011

BACON & THOMAS
625 Slaters Lane, 4th Floor
Alexandria, Virginia 22314

Telephone: (703) 683-0500

S:\Producer\bea\Pending Q...Z\W\WITTMER 534700\appeal brief.wpd

VIII. CLAIMS APPENDIX

1-4. (Canceled)

5. (Rejected) A method for producing software modules for field devices for process automation technology, that encapsulate all the data and functions of the corresponding field devices, wherein the software modules serve as device descriptions and have defined interfaces for application programs in process control systems, comprising the steps of:

generating syntactically correct standard device descriptions from PDM device descriptions, HCF device descriptions or company specific electronic device descriptions for field devices not having a uniform form, or language by means of a first compiler or generator; and

converting the syntactically and semantically correct standard device descriptions further into corresponding software modules by means of a second compiler.

6. (Rejected) The method as claimed in claim 5, wherein:

interfaces and the software modules meet the FDT/DTM specifications.

7. (Rejected) The method as claimed in claim 5, wherein:

the electrical device descriptions are one of: PDM device descriptions, HCF device descriptions or company-specific device descriptions.

8. (Rejected) The method as claimed in claim 5, wherein:

the syntactically and semantically correct standard device

9. (Rejected) The method as claimed in claim 8, further comprising the step of:

producing the EDD 1.1 device descriptions PDM device descriptions

10. (Rejected) The method as claimed in claim 8, further comprising the step of:

using the second compiler to produce graphical user interfaces in XML language from the EDD 1.1 device descriptions.

IX. EVIDENCE APPENDIX

No evidence is submitted herewith.

X. RELATED PROCEEDINGS APPENDIX

No related proceedings have occurred, and none are pending.